



Combinatorial properties of one-dimensional arrangements

Frédéric Cazals

► To cite this version:

Frédéric Cazals. Combinatorial properties of one-dimensional arrangements. *Experimental Mathematics*, 1997, 6 (1). inria-00509987

HAL Id: inria-00509987

<https://inria.hal.science/inria-00509987>

Submitted on 17 Aug 2010

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Combinatorial Properties of One-Dimensional Arrangements

Frédéric Cazals *

IMAGIS-IMAG

BP 53 - 38041 Grenoble cedex 09 - FRANCE

email: Frederic.Cazals@imag.fr

Abstract

Arrangements are an omni-present topic in computational geometry, since many problems in computer graphics and robotics reduce to the study of such sets. Motivated by two problems from these areas —more precisely from ray-tracing and assembly planning, we study in this paper the combinatorial structure of arrangements of segments on a line and of cones on a circle. We show that the numbers of such arrangements are respectively $1.3.5 \dots (2n-1)$ and $(2n)!/n!$, that the probabilities for the i^{th} vertex of a random arrangement to be a beginning point are $1-(i-1)/(2n-1)$ and $1/2$, and that the average numbers of segments or cones the i^{th} vertex is contained in are $(1-i)(i-2n)/(2n-1)$ and $(n-1)/2$. In addition to providing results for the analysis of the ray tracing related and assembly sequencing problems, the constructions used to prove these results provide sampling schemes for generating random inputs usable to test and validate the correctness of programs manipulating arrangements.

Along with the derivation of these identities, we also point out connections between arrangements, sub-diagonal random walks and the ballot problem, as well as other integer sequences.

Topics:Combinatorics, Computational Geometry, Algorithms and Data Structures.

1 Introduction

1.1 Assembly sequencing and arrangements

Assembly sequencing is a domain of robotics the purpose of which is, given a collection of mechanical parts and a class of motions these parts can be moved by, to compute a way (if any) to get the single parts from the whole assembly. Analyzing the disassembly sequences of a given assembly can lead to substantial improvements in many departments. For example to check that the product is disassemblable, or to enable the designers to make sure that the parts which may be serviced often are easily accessible, or to anticipate a better recycling by clustering parts of the same material. Of major practical interest, it also turns out that assembly sequencing is a difficult algorithmic problem since it is in its more general form intractable —see e.g. [Natarajan, 1988]. Nevertheless, restricted yet interesting versions of the problem have been shown to have polynomial time and space algorithms.

For example, let us consider the case of planar polygonal assemblies where the only class of motions allowed is infinite translations and where each split results in two sub-assemblies —[Wilson and Latombe, 1994, Latombe et al., 1996]. The space of motions is described by the circle S^1 since a translation corresponds to a unit vector in the plane. Given any two parts, the set of directions along which one can be translated without colliding the other is described by a cone on the circle S^1 . This is the well known Minkowski difference depicted on figure 1(b) ([Latombe, 1991]). The blocking relations for all the pairs of parts are thus described by $\binom{n}{2}$ cones. Put together, this *arrangement of cones* partitions S^1 into vertices and open cells as shown on figure 2(a). This arrangement is called the *Non-Directional Blocking Graph* or *NDBG* since it gives the blocking relations for any pair of parts and any direction. To each vertex of the arrangement corresponds a directed graph called *Directional Blocking Graph* with a vertex for each part and an edge between vertices i and j if part i collides part j when translated along this direction. A topological sorting of the strong connected components of this graph gives the removable sub-assemblies

*Work on this paper was accomplished while the author was visiting the Robotics Laboratory, Department of Computer Science, Stanford University, Stanford, CA 94305 USA

along this direction. Starting with the full assembly, the disassembly algorithm consists in recursively removing translatable sub-assemblies with the previous scheme.

Performing a worst-case analysis of this algorithm is pretty easy. Indeed, the *NDBG* has at most $O(n^2)$ vertices and each *DBG* has size $O(n^2)$, which gives a space requirement of $O(n^4)$. As for the time complexity of the recursive disassembly, it is $O(n^5)$ since there are at most n levels of recursion, and that each level requires examining at most $O(n^2)$ *DBGs* for which the reduced graph (graph of the strongly connected components) and a topological sorting have to be computed.

On the other hand, coming up with an average-case analysis is much more challenging. Firstly, a precise understanding of the combinatorics of cone arrangements is required. Secondly, some random graph structure is needed for the directional blocking graphs. The latter question is difficult since the number of edges of a *DBG* depends on the geometric information encoded in the relative position of the pairs of parts, which require some definition of random assemblies. This goes beyond the scope of this paper and we skip it. On the opposite, the former problem is more well defined and raises precise questions such as the generation of a random arrangement — see also [Zimmermann, 1994], the probability for a given vertex to be a beginning point or an endpoint of a cone, the average number of cones a given vertex of an arrangement is contained in, etc. These questions are addressed in section 3.

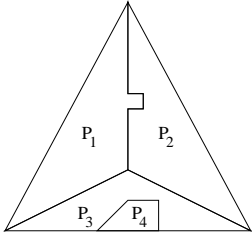
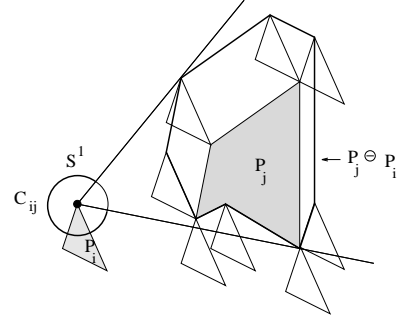


Figure 1: (a) A simple assembly



(b) The Minkowski difference

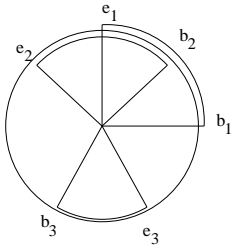
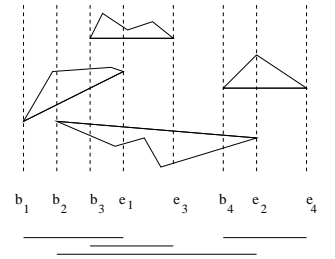


Figure 2: (a) Arrangement of cones on S^1



(b) Arrangement of line segments

1.2 Ray tracing and clustering

Ray tracing is a technique from computer graphics that consists in computing views of scenes defined by geometric primitives. Very often these primitives are polygons defined by their geometry and color, a given object of the scene being defined by a set of such polygons. As an example, consider fig. 3(a) where the whole kitchen model consists of about 25,000 polygons, and objects such as the bowl on the table or the teapot are made of about 1000 polygons. To sketch the ray-tracing algorithm — see [Foley et al., 1990] for the details, let a ray be defined by a point and a direction in $3D$. Rays are used to simulate the light received by the observer's eye, so that the key operation of the whole algorithm consists in finding, for a given ray, the closest object hit in order to plot the corresponding color on the screen of the computer where the algorithm is run.

Reducing the number of ray-polygon intersection tests has ever been a challenging issue. The main paradigm consists in partitioning the volume containing the scene in order to test for intersection those polygons only stored in the voxels of the partition crossed by the ray of interest. An example of such partitioning, the so called uniform grid, is done with a $3D$ axis aligned grid — see [Cazals et al., 1995]

for a discussion of grid-like data structures. The problem of this approach is that whenever too many polygons fall in the same voxel, the spatial partitioning does not result in data partitioning so that the number of ray-polygon intersection tests is not reduced significantly. To remedy this problem, it was observed in [Cazals et al., 1995] that using uniform grids for densely populated areas of the scene called clusters could partially solve the problem. Examples of clusters are the neighborhoods of the bowl, teapot, or door knobs, and are depicted on fig. 3(a)(b). More precisely, a cluster is defined as a subset of objects whose projection along the three axis x, y and z is almost-connected. And since the projection of a polygon on a line is a line-segment, the clustering algorithm analysis turns out to be closely related to the combinatorics of line-segments arrangement — see fig. 2(b). Also, the results presented thereafter in section 2.3 were recently used in [Cazals and Sbert, 1997] in conjunction with integral geometry techniques to define statistics aiming at characterizing standard scenes types such as natural models, architectural scenes, etc.

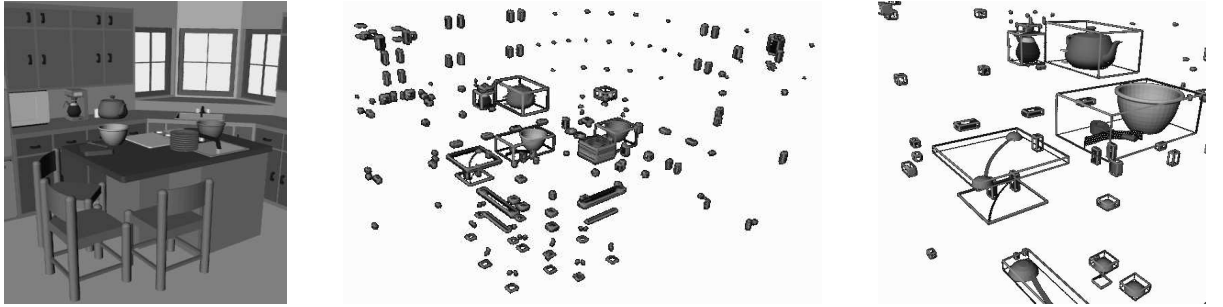


Figure 3: (a)Kitchen model

(b)Clusters: far view

(c)Clusters: close view

1.3 Notations, previous work and results

In the circular and linear cases just described, since we are only interested in the topological properties of the arrangements, we map the $2n$ extremities of the n cones or segments of interest onto the integers $1, 2, 3, \dots, 2n$, which we note $1..2n$. Let S_n and R_n be the sets of all the arrangements of segments and cones of size n , and s_n and r_n their cardinalities. For every vertex of the arrangement, the statistics we are interested in are the following:

- its probability to be a beginning point or an end point,
- its *overlap number*, defined as the average of the number of cones or line segments it is contained in.

The cone and the segment starting at index i and ending at index j are noted (i, j) and $[i, j]$ respectively. Also, the beginning and ending points of a cone or a segment are labeled B and E. For a particular arrangement $a \in R_n$ or S_n and $i \in 1..2n$, we denote by $a[i]$ the operation that returns the letter B or E stored at slot i (this is correct since we suppose that the segments extremities do not overlap). The symbol 1_x with x a boolean expression is the indicator of x , i.e. takes the value 1 if x is true, 0 else.

At last, for a given slot, the number of B letters, the number of E letters and the overlap number are defined as follows:

$$\beta_i^{(n)} = \sum_{a \in S_n} 1_{a[i]=B}, \quad \varepsilon_i^{(n)} = \sum_{a \in S_n} 1_{a[i]=E}, \quad \tau_i^{(n)} = \sum_{\substack{a \in S_n \\ s \in a}} 1_{i \in s}.$$

The corresponding vectors for all the slots $1..2n$ are respectively $\vec{\varepsilon}_n, \vec{\beta}_n, \vec{\tau}_n$. For $n = 2$ e.g., we have (figure 2(b)): $\vec{\beta}_2 = [3, 2, 1, 0]$, $\vec{\varepsilon}_2 = [0, 1, 2, 3]$, $\vec{\tau}_2 = [0, 2, 2, 0]$.

Before proceeding, let us review the previous work. Actually, the numbers s_n and r_n have already appeared in the literature under several forms, in particular in the work of [Touchard, 1950] and [Riordan, 1975] related to the folding-stamps problems. In [Touchard, 1950] [section 4], it is shown that $s_n = 1.3 \dots (2n - 1)$. In [Riordan, 1975], it is stated that the number of pairings of $2n$ points on a circle is also $1.3 \dots (2n - 1)$, and the figure of page 216 –reproduced on the bottom part of our figure 4– shows a mapping from an arrangement on the circle to an arrangement on the line. Indeed, consider a labeling of the points on the circle, say in counterclockwise order. Any pairing (i, j) on the

circle corresponds to the line segment $[i, j]$ on the line, and conversely. At last, a look in the very nice book [Sloane and Plouffe, 1995] shows that sequence s_n has also long been known in relation with the expression of Wallis integrals.

An interesting note from [Riordan, 1975] is the relation between the number of pairings on a circle and the Catalan numbers: pairings where the chords are not allowed to intersect give rise to the Catalan numbers $C_n = \binom{2n}{n}/(2n+1)$, while pairings with crossings between the chords lead to s_n . The author also cites a correspondence between the Catalan numbers and the ballot problems also known as the sub-diagonal random walks problem ([Comtet, 1974, Yaglom and Yaglom, 1964, Knuth, 1973]). Apart from a detailed analysis of the statistics listed above, an interested result of this paper is to show that exactly as in the case of the Catalan sequence C_n , the sequences s_n and r_n can also be viewed in terms of random walks.

The overview of the paper is the following. In section 2 and 3 respectively we study the planar and circular cases. In section 4 we conclude and list interesting problems.

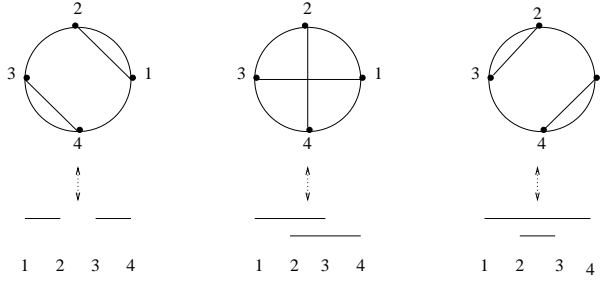


Figure 4: Pairings and arrangements

2 Planar case

2.1 Number of arrangements

As indicated in section 1, the number of line-segment arrangements is given by $s_n = (2n-1)(2n-3)\dots 3.1$. We first give a direct proof of this formula based on an inductive construction of all arrangements of size n . This construction will be used in section 2.2. We then establish another expression for the number s_n based on a sub-diagonal random walk.

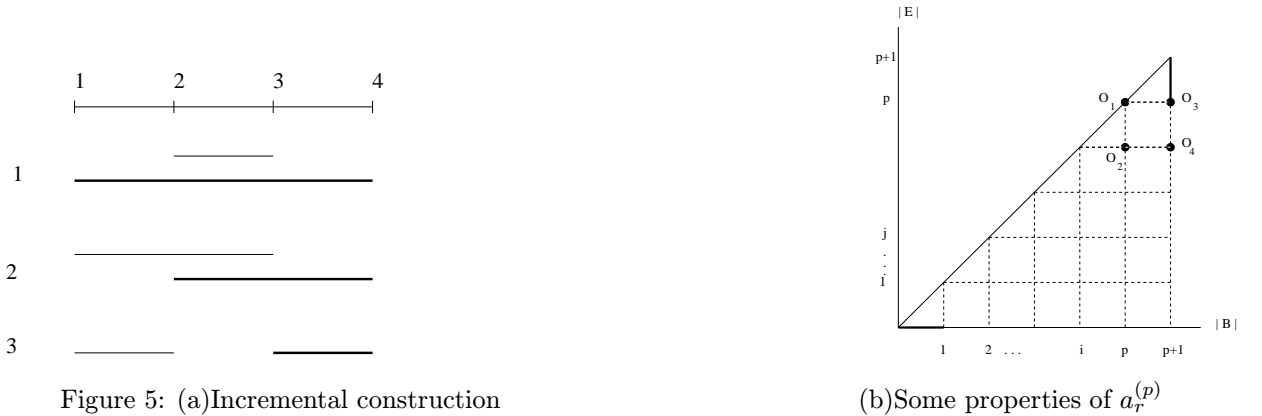


Figure 5: (a) Incremental construction

(b) Some properties of $a_r^{(p)}$

2.1.1 Incremental construction of the arrangements

Suppose we are given the set S_n of all arrangements of size n and that we want to construct S_{n+1} from it. We just have to find a match for the endpoint located at $2n+2$. Let $i \in 1..2n+1$ be this match. If $i = 2n+1$, the newly added segment and the n previous segments do not overlap. If $i < 2n+1$, we are left $2n$ slots in $1..2n+1$. But since the configurations with n segments were defined in $1..2n$, to embed any of these configurations in $1..2n+1 - \{i\}$, we just have to shift by one unit to the right the extremities

of the segments whose index is $\geq i$. We thus have $s_{n+1} = (2n+1)s_n$. For an example, see figure 5(a) where the newly added segment is depicted in bold.

2.1.2 Arrangements and random walks

As already noticed, a segment can be viewed as two endpoints B and E. Constructing an arrangement of size n therefore consists in assigning B and E letters to the $2n$ slots and pairing each E with a B. This process can be seen as a random walk where at stage i in $1..2n$ an horizontal (vertical) move consists in putting a B (E) on slot i . But putting a E consists in closing a segment which means that some letter B must be available at a location $1 \leq k < i$. Moreover, if j represents the number of such letters, the new E can be matched with any of them.

For an arrangement of size n , let $\varphi_n(i, j)$ the function equal to the number of ways we can be left with j unmatched letters B after using i of them, with $1 \leq i \leq n$, $0 \leq j \leq i$. Since the first letter has to be a B, we start at $\varphi_n(1, 1)$ and the total number of arrangements sought is $\varphi_n(n, 0)$. But according to the above discussion, φ satisfies the recurrence

$$\varphi_n(i, j) = \varphi_n(i+1, j+1) + j\varphi_n(i, j-1) \text{ with } 1 \leq i \leq n, 0 \leq j \leq i \quad (1)$$

Solving this recurrence is not that easy. The so-called *reflection principle* of André –[Comtet, 1974]– does not apply since at any time, we have to take into account the past history of the random walk. Also, the general machinery of bivariate generating functions –[Knuth, 1973] exercise 2.2.1.4 and its solution– is of little help. Indeed, let

$$G(x, z) = \sum_{i \geq 0, j \geq 0} \varphi_{i,j} x^i z^j \text{ and } g(x) = G(x, 0) = \sum_{i \geq 0} \varphi_{i,0} x^i$$

Observing that $\varphi_{0,j} = \delta_{0,j}$ ($\delta_{.,.}$ being the Kronecker symbol), the above recurrence can be rewritten as follows using these generating functions:

$$G(x, z) = \frac{1}{xz} (G(x, z) - g(x)) + zG(x, z) + z^2 \frac{\partial G(x, z)}{\partial z}$$

which unfortunately does not give a straightforward asymptotic expansion of $G(x, z)$. We therefore derive a more *ad-hoc* solution.

Theorem 1 *The total number of configurations can be expressed as follows,*

$$s_n = \sum_{r=1}^n a_r^{(n)} \varphi_n(n, r) = \sum_{r=1}^n a_r^{(n)} r! \quad (2)$$

where $\{a_r^{(p)}\}_{p=1..n, r=1..p}$ is a sequence of positive natural integers satisfying

$$a_j^{(p+1)} = a_{j-1}^{(p)} + \sum_{r=j}^p a_r^{(p)} r^{\overline{r-j+1}}, \text{ with } a_1^{(1)} = 1 \text{ and } a_0^{(\cdot)} = 0 \quad (3)$$

and $i^{\overline{j}}$ is the falling factorial $i(i-1) \cdots (i-j+1)$.

Proof: To see that $s_n = \sum_{r=1}^n a_r^{(n)} \varphi_n(n, r)$, observe that we can classify all the arrangements on the numbers of E letters they are finishing by, which is either $1, 2, \dots, n$. And $\varphi_n(n, r) = r!$ since there are respectively $r, r-1, \dots, 1$ choices to close the $1^{st}, 2^{nd}, \dots, r^{st}$ segments corresponding to the r letters E left. We therefore address the computation of the numbers $\{a_r^{(p)}\}_{p=1..n, r=1..p}$.

First observe that at the rank $p+1$, to compute $s_{p+1} = \sum_{r=1}^{p+1} a_r^{(p+1)} \varphi_{p+1}(p+1, r)$ one must beforehand have computed the sequence $\{a_r^{(p)}\}_{r=1..p}$. Indeed, before having used $p+1$ letters B, one must have used p of them. Therefore,

$$\varphi_{p+1}(1, 1) = a_1^{(p)} \varphi_{p+1}(p, 1) + a_2^{(p)} \varphi_{p+1}(p, 2) + \dots + a_p^{(p)} \varphi_{p+1}(p, p)$$

But a straightforward computation leads to

$$\varphi_{p+1}(p, j) = \varphi_{p+1}(p+1, j+1) + \sum_{r=1}^j j^r \varphi_{p+1}(p+1, j+1-r), \quad (4)$$

that is

$$\varphi_{p+1}(1, 1) = \sum_{j=1}^p a_j^{(p)} \left[\varphi_{p+1}(p+1, j+1) + \sum_{r=1}^j j^r \varphi_{p+1}(p+1, j+1-r) \right] \quad (5)$$

Expanding and diagonally rearranging the terms yields to theorem 1. \square

The table below shows the values of the previous sequences up to $n = 10$:

p	$\{a_r^{(p)}\}$										s_p
1	1										1
2	1	1									3
3	3	3	1								15
4	15	15	6	1							105
5	105	105	45	10	1						945
6	945	945	420	105	15	1					10395
7	10395	10395	4725	1260	210	21	1				135135
8	135135	135135	62370	17325	3150	378	28	1			2027025
9	2027025	2027025	945945	270270	51975	6930	630	36	1		34459425
10	34459425	34459425	16216200	4729725	945945	135135	13860	990	45	1	654729075

Figure 6: Numerical values up to $n = 10$

Some interesting facts are the following:

1. From the above recurrence, s_n is obviously super-factorial since $n!$ is just the number of ways to close the n opened segments when the first n letters used are B,
2. We have $a_1^{(p+1)} = s_p$. Indeed, on figure 5(b), the number of ways to reach point O_3 is exactly the number of ways to reach point O_1 : while at the former the only possibility we have is to use a letter B,
3. Also, $a_1^{(\cdot)} = a_2^{(\cdot)}$. On figure 5(b), this means that each path that joins O_2 gives rise to a path to O_3 and a path to O_4 ,
4. From [Sloane and Plouffe, 1995], it appears that the sequence $\{s_p\}_{p=1,2,3,\dots}$ (M3002) has long been known as the 'sequence of double factorials' introduced to make easier the expression of Wallis integrals.

2.2 Beginning points and Endpoints

For any arrangement, the first letter must be a B while the last must be a E. The following theorem actually proves that there is a continuous variation over the indices of the vertices:

Theorem 2 *The number $\beta_i^{(n+1)} = \sum_{a \in S_{n+1}} 1_{a[i]=B}$ satisfies the recurrence relation*

$$\forall i \in 1..2n+1 \quad \beta_i^{(n+1)} = (i-1)\beta_{i-1}^{(n)} + s_n + (2n+1-i)\beta_i^{(n)}, \text{ and } \beta_{2n+2}^{(n+1)} = 0 \quad (6)$$

and the corresponding closed forms are

$$\forall i \in 1..2n \quad \begin{cases} \beta_i^{(n)} = (2n-i)s_{n-1} = s_n - (i-1)s_{n-1} \\ \varepsilon_i^{(n)} = s_n - \beta_i^{(n)} = (i-1)s_{n-1} \end{cases} \quad (7)$$

The probability for the i^{th} vertex of a random arrangement to be a beginning point is therefore

$$1 - (i-1)/(2n-1)$$

Proof: To compute $\beta_i^{(n+1)}$, let us consider the construction of section 2.1.1: once we have fixed the slot $i \in 1..2n+1$ to be matched by the E located at $2n+2$, the number of letters B located in S_n at the indices $1..2n$ are reported on $1..i-1 \cup i+1..2n+1$, and of course, the slot i is credited of s_n letters B. Thus, we have

$$\begin{aligned} \vec{\beta}_{n+1} = & \\ & \begin{bmatrix} s_n & \beta_1^{(n)} & \beta_2^{(n)} & \dots & \beta_{2n}^{(n)} & 0 & \end{bmatrix} & i=1 \\ + & \begin{bmatrix} \beta_1^{(n)} & s_n & \beta_2^{(n)} & \dots & \beta_{2n}^{(n)} & 0 & \end{bmatrix} & i=2 \\ & \vdots \\ + & \begin{bmatrix} \beta_1^{(n)} & \beta_2^{(n)} & \beta_3^{(n)} & \dots & s_n & 0 & \end{bmatrix} & i=2n+1 \end{aligned}$$

The summation column by column gives equation (6). To get the closed form of $\beta_i^{(n)}$, let's set up the recurrence hypothesis $\beta_i^{(n)} = s_{n-1}(2n-i)$ for $i = 1..2n$. With equation (6), we get for any $i \in 1..2n+1$ (notice that the indexes of $\beta_i^{(n)}$ are violated for $\beta_0^{(n)}$ and $\beta_{2n+1}^{(n)}$, but it does not matter since these terms are multiplied by null factors)

$$\begin{aligned} \beta_i^{(n+1)} &= (i-1)s_{n-1}(2n-i+1) + s_n + (2n-i+1)s_{n-1}(2n-i) \\ &= s_n + s_{n-1}(2n-1)(2n-i+1) \end{aligned} \quad (8)$$

But $s_n = s_{n-1}(2n-1)$, which completes the proof for $i = 1..2n+1$. The case $i = 2n+2$ is trivial. As for $\varepsilon_i^{(n)}$, we of course have $\beta_i^{(n)} + \varepsilon_i^{(n)} = s_n$.

As for the probability for the i^{th} vertex of a random arrangement to be a beginning point, it's the previous value divided by the number of arrangements, which is $1 - (i-1)/(2n-1) \square$

2.3 Overlap number

For a given vertex, this number is defined as the average over all the possible arrangements of the number of segments it is contained in. We have the following theorem:

Theorem 3 The number $\tau_i^{(n)} = \sum_{\substack{a \in S_n \\ s \in a}} 1_{i \in s}$ satisfies the recurrence relation

$$\forall i \in 1..2n+1 \quad \tau_i^{(n+1)} = (i-1)(\tau_{i-1}^{(n)} + s_n) + \varepsilon_i^{(n)} + (2n+2-i)\tau_i^{(n)} \text{ and } \tau_{2n+2}^{(n+1)} = 0 \quad (9)$$

More precisely, $\forall i \in 1..2n \quad \tau_i^{(n)} = (1-i)(i-2n)s_{n-1}$.

Thus, the average overlap number of the i^{th} vertex is $\frac{(1-i)(i-2n)}{2n-1}$.

Proof: Here again, we start with a recurrence coming from the incremental construction of the arrangements. We have seen that fixing the slot to be matched by the letter E located at $2n+2$, say the i^{th} one, was giving rise to s_n configurations. See figure 7 for the case $n+1=3$ and $i=2$. In fact, there are three different cases:

- if $i = 1$, all the vertices of the segments from S_n embedded in $2..2n+1$ are covered by the newly added segment,
- if $i = 2n+1$, neither of the segments from S_n and the newly added segment overlap,
- the tricky case is if $i > 1$ and $i < 2n+1$. As we have seen, all the vertices of segments from S_n the index of which were $\geq i$ are shifted of one unit on the right side. Therefore, index i is not only covered by the segments from S_n that were covering it, but is also covered by all the segments of S_n that were ending at index i . On figure 7 e.g., vertex 2 is now covered by the first segment of S_2 in the configuration 1.

We thus have the equality

$$\begin{aligned} \vec{\tau}_{n+1} = & \\ & \begin{bmatrix} 0 & \tau_1^{(n)} + s_n & \tau_2^{(n)} + s_n & \dots & \tau_{2n-1}^{(n)} + s_n & \tau_{2n}^{(n)} + s_n & 0 & \end{bmatrix} & i = 1 \\ + & \begin{bmatrix} \tau_1^{(n)} & \tau_2^{(n)} + \varepsilon_2^{(n)} & \tau_2^{(n)} + s_n & \dots & \tau_{2n-1}^{(n)} + s_n & \tau_{2n}^{(n)} + s_n & 0 & \end{bmatrix} & i = 2 \\ & \vdots \\ + & \begin{bmatrix} \tau_1^{(n)} & \tau_2^{(n)} & \tau_3^{(n)} & \dots & \tau_{2n}^{(n)} & 0 & 0 & \end{bmatrix} & i = 2n+1 \end{aligned}$$

and the summation column by column gives equation (9). Let $\tau_i^{(n)} = (1-i)(i-2n)s_{n-1}$ be the recurrence hypothesis. Using equation (9) we get (transitions from line 1 to 2 and 2 to 3 corresponding to $s_n = (2n-1)s_{n-1}$)

$$\begin{aligned}\tau_i^{(n+1)} &= (1-i)[s_n + (2-i)(i-1-2n)s_{n-1} + (i-1)s_{n-1} + (2n+2-i)(1-i)(i-2n)s_{n-1}] \\ &= (i-1)s_{n-1}[(2n-1)(2n+2-i)] \\ &= (1-i)(i-2n-2)s_n\end{aligned}\tag{10}$$

which proves the theorem. \square

It is interesting to note that this theorem, which was proved using the mechanical inductive construction of the arrangements, also admits a nicer direct proof:

Proof: The slot i is covered by segments of the form $[j, k]$ for $j = 1..i-1$ and $k = i+1..2n$, and there are $(i-1)(2n-i)$ different such segments. But each of these appears exactly s_{n-1} times in the s_n arrangements, since once we have fixed the segment $[j, k]$ we are left an arrangement of $n-1$ segments. \square

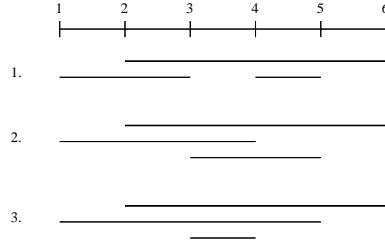


Figure 7: Three configurations of S_3 with slot 2 used

3 Circular case

We now address the questions encompassed in the linear case for the circular one. Actually, due to the equivalent role played by any vertex on S^1 , the situation is easier. We start with the number of arrangements:

Theorem 4 *The number r_n of arrangements of n cones on a circle is equal to $(2n)!/n!$*

Proof: First observe that the set of all possible arrangements can be partitioned into subclasses corresponding to the different pairings of vertices ([Riordan, 1975]), and as recalled above this number is $(2n-1)(2n-3)\dots 3.1$. But the pairing connecting vertices i and j leads to two different cones as indicated on figure 8(a). And since the n pairings are independent, we get $r_n = (2n-1)(2n-3)\dots 3.1.2^n = (2n)!/n!$ \square

The numbers r_n can also be expressed as the result of a random walk as in section 2.1.2. Indeed, the only difference is that when closing a cone, we now have two possibilities as indicated on figure 8(a) instead of one, so that the recurrence is

$$\varphi_n(i, j) = \varphi_n(i+1, j+1) + 2j\varphi_n(i, j-1) \text{ with } 1 \leq i \leq n, 0 \leq j \leq i \tag{11}$$

which we can solve as the linear one.

Also, the classification of the arrangements into pairings solves the question of computing the probability for a given vertex to be a beginning point or an endpoint. Because each pairing gives rise to two configurations as depicted on figure 8(a), these probabilities are equal to $1/2$. As for the overlapping number, we have the following theorem:

Theorem 5 *The average overlap number of a vertex in an arrangement of n cones is $\frac{n-1}{2}$*

Proof: Without loss of generality, let 0 be the index of the vertex we want to compute the overlapping number of, and $1, \dots, 2n-1$ the indices of the subsequent vertices in the counter-clock-wise order. Vertex 0 is covered by cones (i, j) with $i < j$ and $A \in (i, j)$. But vertex i is involved in exactly $i-1$ cones, namely $(i, 1), (i, 2), \dots, (i, i-1)$, which is $\sum_{i=1}^{2n-1} i-1 = (2n-1)(2n-2)/2$. Also cone (i, j) appears exactly r_{n-1} times in the r_n configurations, since the $n-1$ other cones are built from the $2n-2$ remaining vertices. Therefore, the average over all the possible arrangements is $(n-1)(2n-1)r_{n-1}/r_n = (n-1)/2$. \square



Figure 8: (a) Two cones for a pairing

(b) Overlap numbers

4 Conclusions

In this paper we studied some combinatorial aspects of one-dimensional arrangements of segments on a line and of cones on a circle. We showed that the numbers of such arrangements are respectively $1.3.5 \dots (2n-1)$ and $(2n)!/n!$, that the probabilities for the i^{th} vertex of a random arrangement to be a beginning point are $1 - (i-1)/(2n-1)$ and $1/2$, and that the average numbers of segments or cones the i^{th} vertex is contained in are $(1-i)(i-2n)/(2n-1)$ and $(n-1)/2$. The interests of the results are threefold.

Firstly, the analysis in the linear case is of major interest for computer graphics algorithms dealing with objects' projections along lines. Also, the analysis presented about the circular case is the first step for an average case analysis of the *NDBG*-based algorithm computing assembly sequences in the simple case of polygons in the plane moved with infinite translations. Although this particular assembly sequencing problem might appear quite restrictive, it is actually one of the few for which it is reasonable to come up with an implementation for, so that any precise analysis would be of interest. Secondly, from the study of the arrangements combinatorial structure presented in this paper, it is easy to randomly generate such arrangements in order to test and validate geometric software. Thirdly, we pointed out connections between several problems where the sequences s_n and r_n arise, among which arrangements, random walks and ballot problems.

There are still many interesting issues remaining. In particular, getting higher moments for the statistics presented here is an open question. Also, coming up with results in the two-dimensional case would be very interesting, especially since the work done so far deals with arrangements of line in the plane, but not line segments (see [Edelsbrunner, 1986]).

ACKNOWLEDGMENTS: This work received a financial support from Matra Datavision for a joint project on assembly planning with the Stanford University Robotics Laboratory.

The author wishes to thank Danny Halperin and Jean-Marc Schlenker for insightful comments, the reviewer of the first version of this paper for important pointers to the bibliography, and Stéphane Rivière and Ram Ramkumar for rereading the paper.

References

- [Cazals et al., 1995] Cazals, F., Drettakis, G., and Puech, C. (1995). Filtering, clustering and hierarchy construction: a new solution for ray-tracing complex scenes. In *Eurographics'95*, Maastricht. Computer Graphics Forum, Vol. 14 No. 3.
- [Cazals and Sbert, 1997] Cazals, F. and Sbert, M. (1997). Some integral geometry tools to estimate the complexity of 3d scenes. *To appear as INRIA Tech-Report. Also submitted.*
- [Comtet, 1974] Comtet, L. (1974). *Advanced combinatorics*. R. Reidel.

- [Edelsbrunner, 1986] Edelsbrunner, H. (1986). *Algorithms in Combinatorial Geometry*. Springer-Verlag.
- [Foley et al., 1990] Foley, J. et al. (1990). *Computer Graphics: Principles and Practice*. Addison Wesley.
- [Guibas and Seidel, 1987] Guibas, L. and Seidel, R. (1987). Computing convolutions by reciprocal search. *Discrete and computational geometry*, 2:175–193.
- [Knuth, 1973] Knuth, D. (1973). *The art of computer programming - I*. Addison-Wesley.
- [Latombe, 1991] Latombe, J. (1991). *Robot motion planning*. Kluwer.
- [Latombe et al., 1996] Latombe, J., Wilson, R., and Cazals, F. (1996). Assembly sequencing with toleranced parts. *Computer Aided Design*, To appear.
- [Messerve, 1948] Messerve, B. (1948). Double factorials. *American Mathematical Monthly*, 55:425–426.
- [Natarajan, 1988] Natarajan, B. (1988). On planning assemblies. In *4th Symposium in Computational Geometry*.
- [Riordan, 1975] Riordan, J. (1975). The distribution of crossings chords joining pairs of $2n$ points on a circle. *Mathematics of Computation*, pages 215–222.
- [Sloane and Plouffe, 1995] Sloane, N. and Plouffe, S. (1995). *The Encyclopedia of Integer Sequences*. Academic Press.
- [Touchard, 1950] Touchard, J. (1950). Contribution à l’étude du problème des timbres poste. *Canadian Journal of Mathematics*, 2:385–398.
- [Wilson and Latombe, 1994] Wilson, R. and Latombe, J.-C. (1994). Geometric reasoning about mechanical assembly. *Artificial Intelligence*, 71:371–396.
- [Yaglom and Yaglom, 1964] Yaglom, A. and Yaglom, I. (1964). *Challenging mathematical problems with elementary solutions*. Holden-Day.
- [Zimmermann, 1994] Zimmermann, P. (1994). Gaia: a package for the random generation of combinatorial structures. *Maple Technical Newsletter*, 1(1):1–9.